

UNIVERZA V LJUBLJANI  
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Blaž Kostanjšek

# **Nadzor in krmiljenje zelenih sten**

DIPLOMSKO DELO

UNIVERZITETNI ŠTUDIJSKI PROGRAM PRVE STOPNJE  
RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: doc. dr. Mojca Ciglarič

Ljubljana 2014



Rezultati diplomskega dela so intelektualna lastnina avtorja in Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavljane ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje avtorja, Fakultete za računalništvo in informatiko ter mentorja.



Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Preučite področje sistemov za nadzor in krmiljenje zalivanja in gnojenja rastlin (rastlinjaki, zelene stene). Izpostavite slabosti, zaradi katerih niso primerni za domačo rabo. Nato zasnujte svoj sistem, ki bo te pomanjkljivosti odpravil, bo prijazen do uporabnika in bo deloval na nizkocenovni platformi. Predstavite načrt sistema in predlagajte primerne tehnologije. Izberite ustrezne senzorje ter izdelajte prototipno strojno in programsko opremo. Sistem opišite s tehničnega in uporabniškega vidika in predlagajte možne izboljšave za nadaljnji razvoj.



## IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani Blaž Kostanjšek, z vpisno številko **63100249**, sem avtor diplomskega dela z naslovom:

*Nadzor in krmiljenje zelenih sten*

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom doc. dr. Mojce Ciglarič,
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela
- soglašam z javno objavo elektronske oblike diplomskega dela v zbirki "Dela FRI".

V Ljubljani, dne 8. julija 2014

Podpis avtorja:





*Na tem mestu bi se rad zahvalil mentorici doc. dr. Mojci Ciglarič, ter vsem sodelujočim pri projektu EPSIAE - Saša Saftić, Pedro Kostelec, Tea Dobovšek, Boris Savić in Nik Adžić.*



# Kazalo

Povzetek

Abstract

<b>1</b>	<b>Uvod</b>	<b>1</b>
<b>2</b>	<b>Zelene stene</b>	<b>5</b>
2.1	Analiza obstoječih rešitev . . . . .	6
2.2	Ideja izboljšane rešitve . . . . .	8
<b>3</b>	<b>Uporabljene tehnologije</b>	<b>9</b>
3.1	Raspberry Pi . . . . .	9
3.2	Senzorji . . . . .	13
<b>4</b>	<b>Načrt in izvedba sistema za nadzor in krmiljenje zelenih sten</b>	<b>15</b>
4.1	Funkcionalna specifikacija . . . . .	15
4.2	Načrt arhitekture . . . . .	18
4.3	Opis vezja . . . . .	25
4.4	Namestitev in zagon razvojne ploščice Raspberry Pi . . . . .	28
4.5	Zagonska skripta . . . . .	31
4.6	Program za krmiljenje vezja . . . . .	32
4.7	Komunikacija z uporabniškim vmesnikom . . . . .	32
4.8	Uporabniški vmesnik in uporaba . . . . .	33
<b>5</b>	<b>Sklepne ugotovitve in nadaljnje delo</b>	<b>37</b>



# Povzetek

Zelene stene so v zadnjih letih močan trend na področju zelenih tehnologij. Uporabljajo se tako v arhitekturi, kot tudi pri domači pridelavi hrane.

Kljub vse večji razširjenosti zelenih sten pa so sistemi za njihovo krmiljenje in nadzor precej primitivni, dragi ter zahtevni za uporabo. Namen diplomskega dela je izdelava cenovno ugodnega ter tehnološko dovršenega sistema za krmiljenje in nadzor zelenih sten. Produkt diplomskega dela je zanesljiv sistem, ki omogoča nadzor in krmiljenje zelene stene na daljavo, preko preprostega spletnega uporabniškega vmesnika. Sistem je zgrajen na miniaturnem računalniku Raspberry Pi, na katerega so priklopljeni namenski senzorji, za interakcijo z uporabnikom pa ponuja moderen uporabniški vmesnik.

***Ključne besede:*** zelene stene, Raspberry Pi, krmiljenje zelenih sten, senzorji, nadzor zelenih sten



# Abstract

Among the green technologies green walls are becoming increasingly popular. They are used both in the architecture as well as in home gardening.

Despite the increased prevalence of green walls the systems for their monitoring and control are rather primitive, expensive and difficult to use. The purpose of the thesis is to produce an affordable and technologically sophisticated system for monitoring and control of green walls. The product of the thesis is a reliable system that allows you to monitor and control green walls remotely, through a simple web user interface. The system is built on a miniature computer Raspberry Pi, which is connected to dedicated sensors and offers a modern user interface for interaction with user.

***Keywords:*** green walls, Raspberry Pi, controlling green walls, sensors, monitoring green walls





# Poglavje 1

## Uvod

Diplomsko delo je nadaljevanje projekta, izdelanega v okviru mednarodne študentske izmenjave EPSIAE 2013, ki se je odvijala v Birminghamu aprila 2013, tema izmenjave pa so bile zelene tehnologije. V okviru te izmenjave smo s skupino dodiplomskih študentov univerzitetnega programa računalništvo in informatika na Fakulteti za računalništvo in informatiko razvili idejo o krmiljenju in nadzoru vertikalnih vrtov, imenovanih zelene stene. Pri tem smo se osredotočili predvsem na to, da je sistem poceni, enostaven za uporabo, ter uporablja sodobne tehnologije. Tu smo želeli izpostaviti predvsem enostavnem uporabniški vmesnik, ki je preko interneta dostopen na daljavo in je preko njega možno krmiliti zeleno steno ter spremljati njeno stanje. Sistem bi bil tako primeren predvsem za manjše stene, ki jih imajo uporabniki v notranjih prostorih, kljub temu pa bi zaradi svoje modularnosti omogočal tudi krmiljenje večjih sten. Sama ideja je vključevala tudi zbiranje podatkov v oblaku, kar bi nam kasneje omogočilo s pomočjo umetne inteligence nad temi podatki izvajati različne analize, ter končnemu uporabniku ponuditi priporočilni sistem. Tako bi sistem uporabniku, glede na okolje v katerem se nahaja njegova zelena stena, priporočil katere rastline so primerne, hkrati pa glede na rastline ki jih uporabnik posadi, skušal kar najbolj prilagoditi parametre zelene stene. Poleg same ideje smo razvili tudi enostaven prototip, katerega sem v okviru diplomskega dela funkcionalno ter tehnično izpopolnil.

Sam projekt smo že v osnovi razdelili na smiselne enote, izmed katerih sem sam večino prispeval k načrtovanju in izdelavi vezja, ter k logiki za krmiljenje vezja, svoj del pa sem prispeval tudi k uporabniškemu vmesniku. V diplomskem delu sem se tako posvetil predvsem krmilnemu delu sistema, vseeno pa se na kratko dotaknem tudi uporabniškega vmesnika, saj predstavlja pomemben del celotnega projekta, hkrati pa končnemu uporabniku na prvi pogled prikaže zmogljivosti celotnega sistema. Zbiranje podatkov v oblaku ter uporaba umetne inteligence nad temi podatki nista del diplomskega dela, saj se v okviru tega projekta tej temi nisem posvečal. V okviru projekta EPSIAE izdelan sistem je imel precej pomanjkljivosti, zato je bila najprej potrebna odprava le-teh. Med drugim je bilo potrebno poiskati primernejšo rešitev za merjenje vlage, saj prvotni senzor za naše pogoje ni bil primeren. Prav tako je bilo potrebno poiskati rešitev za merjenje EC vrednosti vode, saj je bila le-ta zgolj ideja prvotnega sistema. V sicer precej dodelanem uporabniškem vmesniku pa je bilo potrebno na novo zasnovati prikaz meritev, saj je bil prikaz meritev pri veliki količini podatkov praktično neuporaben. Poleg tehničnih težav pa smo se pri prvotnem sistemu srečali tudi z drugimi težavami; tako je bilo za delovanje sistema potrebno preučiti različne mešanice substratov ter izbrati primerne za zeleno steno, temu primerno pa tudi prilagoditi meritve vlažnosti. Ker pa tema diplomskega dela obsega zgolj krmiljenje in nadzor zelenih sten, bom takšne probleme le na kratko omenil.

Marsikateri del prvotnega prototipa se je med izdelavo diplomskega dela precej izpopolnjeval ali celo povsem spremenil, v diplomskem delu pa je zgolj opisan sistem tak kakršen je ob končanem diplomskem delu.

V diplomskem delu sprva predstavim zelene stene, njihovo zgradbo ter dosedanje načine krmiljenja. V nadaljevanju predstavim problematiko obstoječih sistemov, ter predlagam izboljšavo obstoječih sistemov. V 3. poglavju na kratko opišem tehnologije uporabljene pri sami implementaciji, ki so ključnega pomena za razumevanje delovanja celotnega sistema. V 4. poglavju sledi obsežen opis dejanske implementacije sistema, v zadnjem po-

glavju pa sledi pregled delovanja sistema, problemi takšne implementacije ter predlogi za izboljšave ter nadaljnje delo.



## Poglavje 2

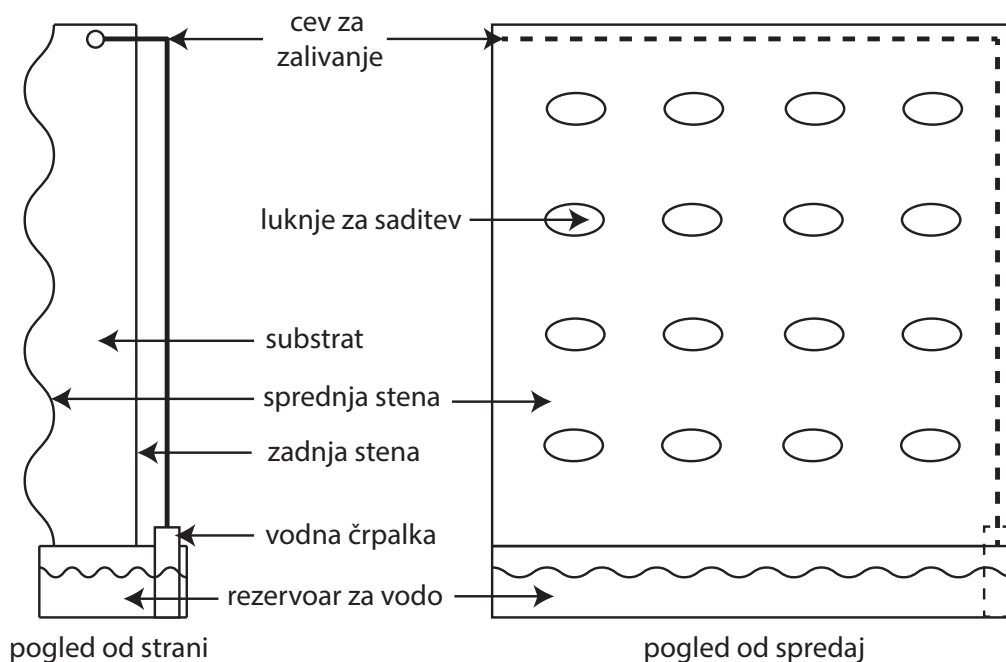
### Zelene stene

Zelene stene, ali tudi "žive stene", kot jih nekateri imenujejo, so v zadnjih letih postale trend tako na področju arhitekture, kot tudi na področju zelenih tehnologij, skrbi za okolje ter doma pridelane hrane.

Zelene stene so zgolj vrtovi v vertikalni obliki, pa vendar ravno zaradi vertikalne postavitve omogočajo umeščanje vrtov v okolja, kjer za običajne vrtove ni prostora. Predvsem so to gosto poseljena mesta, skoraj brez zelenih površin. Zelene stene so tako idealen način za ozelenitev mest, saj jih je možno zgraditi na zunanjih stenah poslopij. V gosto poseljenih mestih pa lahko zelene stene igrajo tudi vlogo vrtov za domačo pridelavo hrane, saj predvsem prebivalci stanovanjskih blokov nimajo svojih vrtov. Zelene stene rešijo ta problem, saj lahko uporabnik vertikalni vrt postavi kar v stanovanju oziroma na balkonu.

Ozelenitev ter gojenje rastlin na mestih, kjer na klasičen način le-to ni mogoče, je zagotovo največja prednost uporabe zelenih sten, a še zdaleč ne edina. Poleg ozelenitve so zelene stene hkrati tudi odlična toplotna izolacija gradbenih objektov, kar posledično vodi k zniževanju stroškov za ogrevanje ter hlajenje objektov[3]. Prispevajo tudi k zmanjševanju onesnaženosti zraka ter izboljšujejo kvaliteto zraka notranjih prostorov [1].

V primerjavi s klasičnimi vrtovi zelene stene za svojo rast porabijo tudi manj vode, saj so v veliki večini zgrajene tako, da ves čas znotraj sistema



Slika 2.1: Običajna zgradba zelene stene.

kroži ista voda - iz rezervoarja se dovaja na vrh stene, od koder po notranjosti stene steka nazaj v rezervoar. Potrebno je zgolj dolivanje porabljene vode.

## 2.1 Analiza obstoječih rešitev

### 2.1.1 Zgradba

Po svetu so zelene stene razširjene v različnih oblikah, v osnovi pa so vse zgrajene iz dveh plošč, med katerima se nahaja substrat, tako kot to prikazuje slika 2.1. Po zgradbi jih vseeno delimo glede na to, na kakšen način, oziroma iz katerih materialov so sestavljene. Najbolj pogosta načina izdelave sta:

**Mehka sprednja stena** Pri tem načinu izdelave je sprednja stena prekrita z mehkim materialom, po katerem se rastline lahko razraščajo, najpogosteje se v ta namen uporabljata filc in klobučevina. Prednost tega

načina izdelave je v tem, da rastline lahko rastejo na celotni površini, in ne zgolj na vnaprej določenih in pripravljenih mestih. Slabost takšnega sistema pa je v tem, da sprednja stena ni povsem vodoodporna in je možno uhajanje vode, kar pa lahko predstavlja problem pri postavitvah v notranjih prostorih. Vsekakor pri zunanjih stenah to ni problem.

**Trda sprednja stena** Pri tem načinu izdelava pa je sprednja stena prekrita s trdim materialom, ki ne prepušča vode. Rastline lahko rastejo le v vnaprej pripravljenih odprtinah (glej sliko 2.1), stena pa na pogled spominja na več skupaj zloženih lončkov. Prednost tega sistema je v tem, da je voda vedno znotraj zelene stene, kar je zelo zaželeno pri postavitvah v notranjih prostorih.

### 2.1.2 Krmiljenje

Pri krmiljenju zelenih sten je proces pravzaprav identičen krmiljenju rastlinjakov, oziroma drugih namakalnih sistemov, ki pa so na trgu prisotni že več desetletij. Predvsem je tu govora o večjih rastlinjakih, namenjenih industriji. Tudi krmiljenja zelenih sten namreč v večini primerov temeljijo na isti tehnologiji, uporabljajo iste senzorje ter namakalne sisteme. Med bolj znanimi sistemi so sistemi podjetij Humko[9], Gsky[10], Ambius[11] ter Bin Fen[12]. Slaba stran vseh teh sistemov pa je predvsem njihova cena ter zahtevnost uporabe.

Običajno imamo v takšnih sistemih centralno krmilno enoto, v katero so povezani senzorji za merjenje vlage. Glede na meritve vlage nato reguliramo namakanje sistema. Večji sistemi omogočajo tudi meritve vsebnosti hranilnih snovi, ter temu primerno doziranje le-teh, vendar pa so takšni sistemi zelo redki, ravno zaradi visoke cene. Upravljanje takšnih sistemov je običajno možno le neposredno na centralni krmilni enoti, spremljanja meritev in časovni pregled stanja sistema pa običajno nimamo. V redkih primerih takšni sistemi omogočajo obveščanje o izrednih dogodkih preko SMS sporočil.

## 2.2 Ideja izboljšane rešitve

Pri snovanju našega sistema smo se osredotočili predvsem na pomanjkljivosti že obstoječih sistemov. Zastavili smo nekaj ključnih lastnosti takšnega sistema:

**Nizka cena** Kljub tehnični dovršenosti sistema, naj bo le-ta cenovno ugoden. S tem je sistem primeren tudi za manjše stene, namenjene osebni uporabi.

**Nadzor na daljavo** Sistem mora imeti možnost upravljanja na daljavo. Uporabniški vmesnik naj bo dostopen preko interneta kar s pomočjo brskalnika, ponuja pa naj časovni pregled stanja stene, spreminjanje nastavitev, ter možnost ročnega izvajanja ukazov.

**Enostavna uporaba** Zagon ter delovanje sistema naj bosta povsem avtomatizirana. Ob prvi uporabi naj se sistem zažene s privzetimi nastavitvami, kasneje naj ima uporabnik možnost spreminjanja nastavitev prek uporabniškega vmesnika. V uporabniškem vmesniku naj bodo vnaprej pripravljene nastavitve za velik nabor rastlin. S tem uporabnik ne potrebuje nobenega predznanja - zgolj izbere nastavitve za zeleno rastlino.

**Modularnost sistema** Sistem naj omogoča enostavno dodajanje/odstranjevanje novih senzorjev, glede na uporabnikove potrebe. Tako imamo lahko sistem, ki zgolj uravnava vlago, ali pa sistem, ki poleg vlage skrbi tudi za vsebnost hranilnih snovi, primerno količino svetlobe, uravnavanje temperature, itd.

Vse izmed zgoraj naštetih lastnosti smo pri izdelanem prototipu tudi upoštevali. Za osnovo krmilnega dela smo si izbrali Raspberry Pi [4], s čimer smo poenostavili reševanje problemov nadzora na daljavo, ter modularnosti sistema.



## Poglavje 3

# Uporabljene tehnologije

V tem poglavju so opisane tehnologije, uporabljene pri razvoju sistema za nadzor in krmiljenje zelene stene. Ker pa je vseh tehnologij preveč, da bi jih tu opisovali, so spodaj opisane zgolj tiste, ki so ključnega pomena za razumevanje diplomskega dela.

### 3.1 Raspberry Pi

Raspberry Pi je izredno poceni računalnik v velikosti kreditne kartice. Razvila ga je organizacija Raspberry Pi Foundation, z namenom promoviranja učenja osnov računalništva v šolah. Gre za običajen računalnik, na katerega lahko priklopimo zaslon, miško ter tipkovnico. Raspberry Pi je dovolj zmogljiv, da je na njem možno brskati po spletu, predvajati videoposnetke visokih ločljivosti, urejati dokumente in preglednice ali pa igrati igrice [5].

Tisto, kar naredi Raspberry Pi uporabnega v takšnih sistemih, kot je naš, pa je njegova sposobnost komuniciranja z zunanjim svetom. Z njim namreč lahko beremo vrednosti različnih senzorjev ter krmilimo druge elektronske naprave, hkrati pa imamo v isti napravi tudi vse lastnosti običajnega računalnika. Ravno ta lastnost pa je tista, zaradi katere je Raspberry Pi najprimernejša izbira za naš projekt. Obstaja namreč veliko razvojnih ploščic, ki ponujajo komunikacijo z zunanjim svetom (med njimi gotovo najbolj po-



Slika 3.1: Raspberry Pi model B. Vir: [www.raspberrypi.org](http://www.raspberrypi.org)

znan sistem Arduino), vendar nobena izmed njih ne omogoča, da imamo na isti napravi nameščen operacijski sistem, ter nam tako sistem služi tudi kot strežnik. Po drugi strani pa je Raspberry Pi tudi cenovno bolj ugoden od alternativnih produktov. Cena naprednejšega modela, ki je za naš sistem bolj primeren, znaša manj kot 40 evrov.

### 3.1.1 Specifikacije

Kot smo že omenili, obstajata dva različna modela Raspberry Pi. Osnovni "A" model, ter naprednejši "B" model. Ker osnovni model nima internetnega priključka, in ker želimo da naš sistem omogoča upravljanje na daljavo preko interneta, si bomo tu ogledali zgolj lastnosti naprednejšega "B" modela.

Velikost računalnika je 85.6mm x 56mm in tehta zgolj 45 gramov. Prikazan je na sliki 3.1. Napajanje poteka preko microUSB vmesnika, najmanjši tok, pri katerem vezje še deluje pa je 700mA. Na ploščici imamo na voljo:

- 512MB pomnilnika (SDRAM)

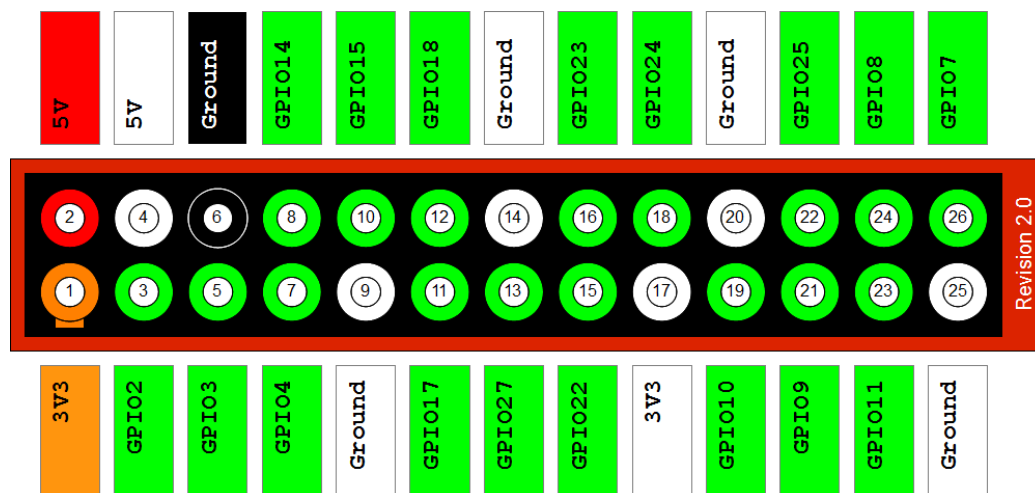
- 700 MHz ARM11 procesor
- 2x USB 2.0
- Ethernet priključek
- reža za spominsko kartico SD
- HDMI izhod
- CSI video vhod
- 3.5mm avdio izhod
- 8x GPIO, I2C, SPI, UART

### 3.1.2 GPIO

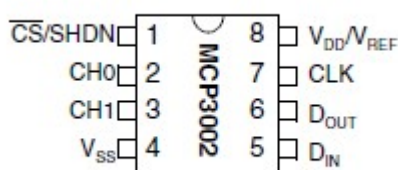
Raspberry Pi nam za komunikacijo z zunanjim svetom ponuja 8 povsem splošno namenskih vhodov/izhodov. Poleg teh so na voljo tudi drugi, ki pa lahko služijo kot I2C, SPI ali UART vmesnik. Na sliki 3.2 je prikazana njihova razporeditev na 26 pinskem konektorju, ki se nahaja neposredno na ploščici.

### 3.1.3 Analogni vhod/izhod

Analognih vhodov/izhodov nam Raspberry Pi v osnovi ne ponuja, jih pa z nekaj dodatnimi elementi brez težav dodamo sami. V ta namen potrebujemo analogno-digitalni pretvornik. V našem primeru smo se odločili za element MCP3002, ki ga prikazuje slika 3.3. Omogoča nam pretvarjanje dveh analognih kanalov v digitalno obliko. V primeru da bi želeli več kanalov, lahko izberemo element MCP3004 ali MCP3008, ki nam ponujata 4 oziroma 8 kanalov. MCP3002 z drugimi napravami komunicira preko SPI vmesnika.



Slika 3.2: Razporeditev kontaktov na GPIO vmesniku. Vir: [ngng.gotovac.org](http://ngng.gotovac.org)



Slika 3.3: Shema analogno-digitalnega pretvornika MCP3002. Vir: [www.raspberrypi.org](http://www.raspberrypi.org)

## 3.2 Senzorji

Za komunikacijo elektronskih naprav z zunanjim svetom imamo različne senzorje, ki z napravami komunicirajo na različne načine. Spodaj so opisani senzorji, ki smo jih uporabili pri implementaciji sistema za nadzor in krmljenje zelene stene.

### 3.2.1 Temperaturni senzor

Za temperaturni senzor smo izbrali senzor DS1820 [7], ki je cenovno ugoden ter dovolj natančen za naše potrebe. Omogoča tudi enostavno branje preko "1-wire" wmesnika. Osnovne značilnosti senzorja so prikazane v tabeli 3.1.

Napajanje	od 3.0V do 5.5V
Območje meritev	od $-55^{\circ}\text{C}$ do $+125^{\circ}\text{C}$
Natančnost meritev	$\pm 0.5^{\circ}\text{C}$
Ohišje	3 TO-92

Tabela 3.1: Značilnosti temperaturnega senzorja DS1820.

### 3.2.2 Senzor vlage

Ker gre v našem primeru za merjenje vlage v substratu, moramo izbrati temu primeren senzor. Običajni senzorji vlage so namreč namenjeni merjenju vlage v zraku. Senzorji, ki so namenjeni merjenju vlage v substratu večinoma delujejo na principu prevodnosti substrata. Ker so takšni senzorji zelo preprosti, smo ga izdelali kar sami. V substrat zarinemo dve prevodni palici, sklenjeni v tokokrog. Glede na upornost med palicama nato določimo vlažnost zemlje. Vlažna zemlja namreč veliko bolje prevaja električni tok kot suha. Raspberry Pi to vrednost bere tako, da s pomočjo analogno-digitalnega pretvornika meri padec napetosti med palicama. S pomočjo dodatnega upora pa določimo razpon meritev, saj imajo različni substrati različne prevodnosti glede na vlažnost. Običajno je ta upor vrednosti med 47K ter 75K,

točno vrednost pa določimo eksperimentalno in sicer tako, da v več posodic s substratom dolijemo različne količine vode, ter nato skušamo dobiti čimbolj natančne meritve glede na količino vode v substratu.

### 3.2.3 EC senzor

Z senzorjem elektro-konduktivnosti merimo količino ionov v raztopini. Glede na to vrednost lahko določimo, kakšna je količina hranilnih snovi, raztopljenih v vodi. Vodovodna voda ima EC vrednost okoli 1. Običajno takšni senzori vračajo analogen signal, ki ga po vnaprej določeni funkciji krivulje, ki jo določi proizvajalec, spremenimo v vrednost EC. Za našo implementacijo smo si izbrali senzor LFS1K0 proizvajalca IST, njegove značilnosti pa so prikazane v tabeli 3.2.

Napajanje	do 1.6V
Območje meritev	od $5\mu S/cm$ do $50mS/cm$
Frekvenca meritev	od 300 do 3000Hz
Temperaturno območje delovanja	$0^{\circ}C$ do $+90^{\circ}C$

Tabela 3.2: Značilnosti EC senzorja LFS1K0.

### 3.2.4 Zaznavanje nivoja vode

Zaznavanje nivoja vode lahko na enostaven način izvedemo s pomočjo več indikatorjev tekočine. Vsak indikator nam pove, ali je na tisti višini tekočina ali ne. Zaznavanje tekočine prav tako lahko izvedeno na zelo enostaven način. Vse kar potrebujemo sta dva vodnika, razmaknjena za nekaj deset milimetrov. Vodnika sklenemo v tokokrog, ter preverjamo ali skozi teče tok. V primeru da sta vodnika v vodi, bo med njima stekel tok, sicer pa ne. Ker pa je ta tok premajhen, da bi ga zaznali na vhodu Raspberry Pi-ja, si moramo pri tem pomagati s tranzistorjem, ki nam ta tok ojača.

## Poglavje 4

# Načrt in izvedba sistema za nadzor in krmiljenje zelenih sten

V tem poglavju je opisana sama implementacija zelene stene. Na začetku je opisana funkcionalna specifikacija, nato pa v naslednjih poglavjih celoten postopek od načrtovanja arhitekture sistema pa vse do navodil za postavitev sistema ter navodila za uporabo.

### 4.1 Funkcionalna specifikacija

Od sistema za nadzor in krmiljenje zelene stene pričakujemo:

**Zalivanje** Sistem mora sam skrbeti za zalivanje/namakanje zelene stene. Za zalivanje zelene stene sistem uporablja vodo iz rezervoarja. Voda se pri zalivanju zelene stene steka nazaj v rezervoar, s čimer dosežemo manjšo porabo vode, rezervoar pa je potreben tudi zaradi gnojenja sistema.

Sistem si pri odločanju o tem kdaj zalivati pomaga s senzorjem za merjenje vlage v substratu, za samo zalivanje pa se uporablja črpalka za vodo, ki nato preko cevi s polprepustno membrano dovaja vodo na vrh zelene stene.

**Gnojenje** Skrb za pravo količino gnojila je ena izmed glavnih zahtev našega sistema. Gre namreč za funkcionalnost, zaradi katere ima naš sistem dodatno vrednost v primerjavi z obstoječimi rešitvami. Večina obstoječih sistemov za nadzor zelenih sten namreč zagotavlja zgolj zalivanje zelenih sten, le redki sistemi pa skrbijo tudi za primerno količino hranilnih snovi v substratu.

Za merjene količine hranilnih snovi v substratu si sistem pomaga z EC senzorjem, ki smo ga opisali v prejšnjem poglavju. Hranilne snovi se nato dozirajo v rezervoar z vodo, od koder se pri zalivanju prenesejo v substrat.

**Polnjenje rezervoarja** Kot smo že omenili sistem uporablja rezervoar za vodo, iz katerega črpa vodo za zalivanje. Potrebno je zagotoviti da vode v rezervoarju ne zmanjka.

Sistem bo meril nivo vode, ter v primeru da bo nivo vode prenizek, odprl elektromagnetni ventil, ki bo omogočil dotok vode iz vodovodnega omrežja. V primeru, kjer vodovodna napeljava do samega sistema ne bo mogoča, bo sistem obvestil skrbnika, da je nivo vode prenizek, nato pa bo moral skrbnik sam poskrbeti za napolnitev rezervoarja.

**Spremljanje meritev in stanja** V vsakem trenutku naj bo na voljo pogled na trenutne meritve, ter stanje vseh delov sistema. Spremljanje stanja naj bo mogoče tudi na daljavo. Ker Raspberry Pi omogoča povezovanje v splet, to ne bo problem.

**Beleženje zgodovine meritev** Vse meritve sistema in obvestila o stanjih naprav naj sistem beleži. Uporabnik naj ima možnost pregleda stanja zelene stene skozi čas.

**Opozorila** V primeru da sistem naleti na kakšno težavo, ali pa da sistem želi o čem obvestiti uporabnika (na primer v zgoraj omenjenem primeru, ko mora uporabnik sam poskrbeti za napolnitev rezervoarja) moramo



poskrbeti da uporabnik prejme opozorilo. Uporabnika lahko sistem opozori na različne načine:

- preko uporabniškega vmesnika,
- preko elektronske pošte,
- preko SMS sporočila ali pa
- enostavno z svetlobnim (LED dioda) ali zvočnim indikatorjem.

**Nastavitve** Sistem naj ponuja možnost spreminjanja nastavitev, ki vplivajo na delovanje sistema. Uporabnik naj ima možnost nastavljanja svojih profilov delovanja, saj različni organizmi potrebujejo različne količine virov.

Sistem lahko hrani tudi svojo zbirko organizmov, s čemer se uporabniku še bolj poenostavi nastavljanje krmiljenja zelene stene.

**Enostavnost uporabe** Sistem lahko sam po sebi deluje še tako dobro, pa vendar, če ga uporabnik ne zna uporabljati, ne koristi kaj dosti. Potrebno je zagotoviti delovanje sistema s kar najmanj začetnimi nastavitvami. Sistem naj se ob prvi uporabi postavi v privzeto stanje, ki naj bo primerno za povprečnega uporabnika. Tako uporabniku ne bo potrebno nastavljanje nastavitev sistema v primeru da ne bo imel posebnih zahtev.

**Stabilnost** Najbolj nujna zahteva takšnega sistema je prav gotovo stabilnost samega sistema. Zagotovljeno mora biti delovanje v vseh možnih pogojih, paziti je potrebno da zajamemo tudi dogodke, do katerih sicer pri pravilni uporabi ne bo prišlo.

Ena izmed bolj pomembnih stvari glede stabilnosti je zagotavljanje ponovne vzpostavitve sistema v primeru izpada električnega napajanja.

**Ročne operacije** Poleg povsem avtomatiziranega krmiljenja zelene stene naj bo na voljo tudi možnost proženja ročnih operacij. To nam pride prav predvsem v primerih ko na primer posadimo nov organizem in

je potrebno takoj po posaditvi organizem dobro zaliti. Sistem sam bi namreč pričel z zalivanjem šele takrat ko bi zaznal da je vlažnost substrata v celotni zeleni steni prenizka.

**Zaznavanje napak** Obvezno je tudi zaznavanje raznih napak, ki se lahko zgodijo med delovanjem sistema. Med te spadajo:

- odpoved senzorjev,
- odpoved delovanja črpalke,
- zamašitev cevi/filtrov,
- odsotnost vode v vodovodnem omrežju,...

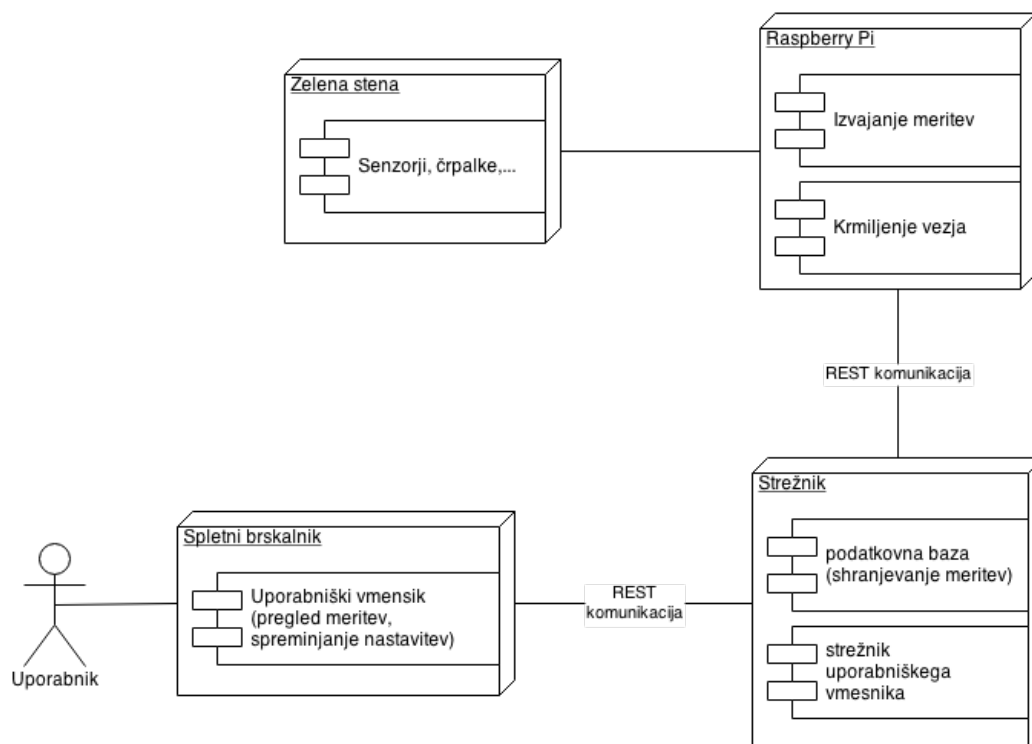
**Priporočila** Ena izmed opsijskih razširitev sistema so priporočila uporabniku. Sistem naj glede na morebitne napačno izbrane uporabnikove nastavitve priporoči primerne nastavitve, ali pa uporabniku zgolj priporoči bolj optimalne nastavitve za izbrane organizme ter okolje v katerem se zelena stena nahaja.

## 4.2 Načrt arhitekture

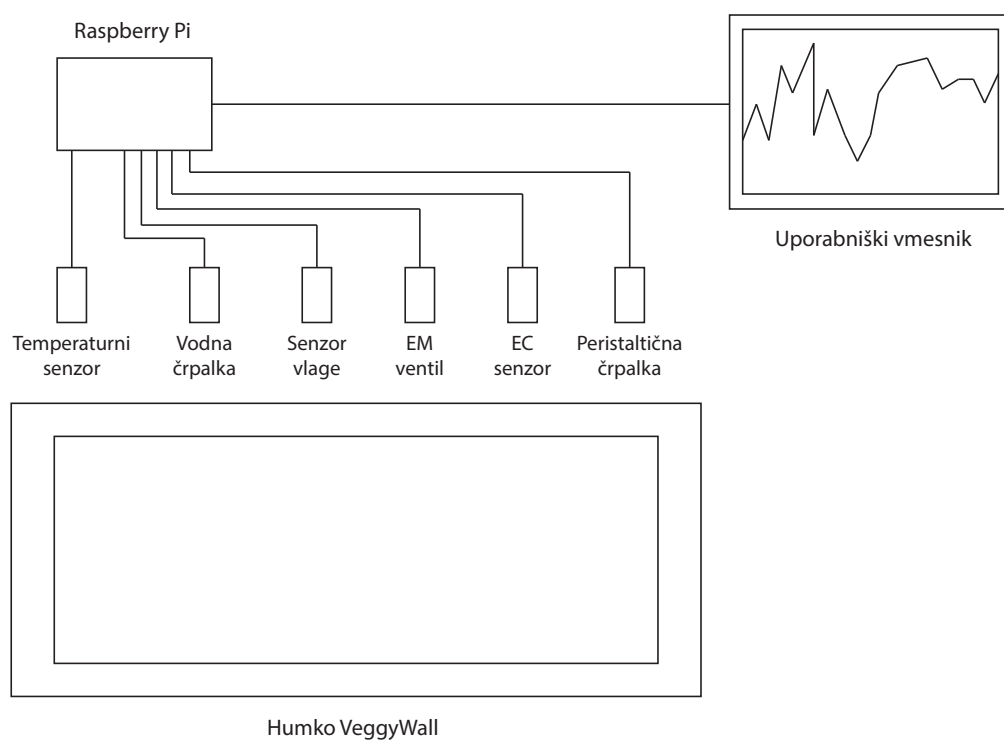
Celoten sistem za nadzor zelene stene je sestavljen iz več enot, ki si jih bomo v tem razdelku podrobneje ogledali. Načrt arhitekture prikazuje slika 4.1, dejanska arhitektura izdelanega sistema pa je prikazana na sliki 4.2. Opazimo lahko, da je vlogo strežnika v našem sistemu prevzel kar Raspberry Pi.

### 4.2.1 Humko VeggyWall

Za samo ogrodje za zeleno steno smo izbrali VeggyWall panel proizvajalca Humko. Gre za manjši, stilsko oblikovan panel, ki je namenjen bolj kot okras notranjih prostorov. Panel je izredno tanek in na zadnji strani spodaj že vsebuje praktičen rezervoar. S tem sam panel že ustreza našim zgoraj opisanim zahtevam. Slika 4.3 prikazuje možnost uporabe takšnega panela.



Slika 4.1: Načrt arhitekture celotnega sistema.



Slika 4.2: Arhitektura celotnega sistema.



Slika 4.3: VeggyWall panel podjetja Humko. Vir: [www.greenwalls.si](http://www.greenwalls.si)

#### 4.2.2 Sistem za zalivanje

Sistem za zalivanje skrbi za štiri glavne procese:

**Merjenje vlažnosti v substratu** se izvaja na več mestih s senzorji vlažnosti opisanimi v prejšnjem poglavju.

**Dovajanje vode na vrh zelene stene** poteka s pomočjo manjše vodne črpalke, ki vodo preko cevi dovaja na vrh zelene stene, kjer se nato preko polprepustne membrane enakomerno razporedi po zgornjem delu zelene stene.

**Merjenje nivoja vode** je pomembno v dveh primerih. V primeru da je v rezervoarju premalo vode za normalno delovanje vodne črpalke, ter v primeru ko želimo sistem obvestiti da je rezervoar poln in naj preneha z dolivanjem vode v rezervoar.

Merjenje nivoja vode je realizirano z več indikatorji tekočine, kakršne smo opisali v prejšnjem poglavju.

**Polnjenje rezervoarja** v primeru nizkega nivoja vode, oziroma obveščanje uporabnika o nizkem nivoju vode v rezervoarju. V primeru da zgolj obvestimo uporabnika, to storimo na enega izmed načinov, ki smo jih opisali med zahtevami, v primeru da rezervoar dejansko napolnimo, pa to storimo tako, da odpremo elektromagnetni ventil, ki povezuje rezervoar z vodovodnim sistemom.

### 4.2.3 Sistem za gnojenje

Sistem za gnojenje zajema dva procesa:

**Merjenje količine hranilnih snovi** se izvaja v rezervoarju, kjer s pomočjo EC senzorja, opisanega v prejšnjem poglavju, merimo konduktivnost v vodi. Pri merjenju moramo biti pozorni na to, da v času doziranja hranilnih snovi meritev ne izvajamo, saj je takrat koncentracija hranilnih snovi v nekaterih predelih rezervoarja večja kot v drugih, in je potrebno počakati da se hranilne snovi enakomerno porazdelijo v vodi. Prav tako moramo biti pozorni na to, da meritev ne izvajamo v primeru praznega rezervoarja, saj meritve ne bodo pravilne, če EC senzor ne bo v vodi.

**Doziranje gnojil** se lahko izvaja na dva načina:

- Pri prvem načinu imamo hranilne snovi v posodah, ki se nahajajo nad rezervoarjem ter imajo na spodnji strani izhod za cev, preko katere poteka doziranje hranilnih snovi. Količino doziranja hranilnih snovi nato reguliramo z odpiranjem in zapiranjem elektromagnetnega ventila, ki se nahaja med posodo in rezervoarjem. Slabost tega načina doziranja je v nenatančnosti odmerjanja količine hranilnih snovi.
- Drugi način doziranja hranilnih snovi je bolj natančen od prvega in za doziranje uporablja peristaltično črpalko. Peristaltična črpalka nam omogoča da količino doziranih hranilnih snovi določimo na mililiter natančno (oziroma natančneje - odvisno od natančnosti

same peristaltične črpalke). Pri tem načinu doziranja tudi nismo omejeni s tem da morajo biti posode s hranilnimi snovmi locirane nad rezervoarjem, saj peristaltične črpalke brez težav tekočino črpajo tudi navzgor.

Oba načina doziranja hranilnih snovi sta prikazana na sliki 4.4. Ne glede na to, kateri način doziranja hranilnih snovi uporabljamo, je z vidika krmilne enote proces doziranja gnojila enak. Oba načina namreč dozirata hranilne snovi tako da digitalni izhodni kanal postavimo na 1, in mirujeta takrat, ko digitalni izhodni kanal postavimo na 0. Jasno pa je, da je potrebno nastaviti, koliko časa poteka doziranje hranilnih snovi od vsaki zahtevi za gnojenje. Ta pa ni odvisna zgolj od načina doziranja, temveč tudi od izbire samih črpalk, cevi, elektromagnetnih ventilov, gostote hranilnih snovi, itd.

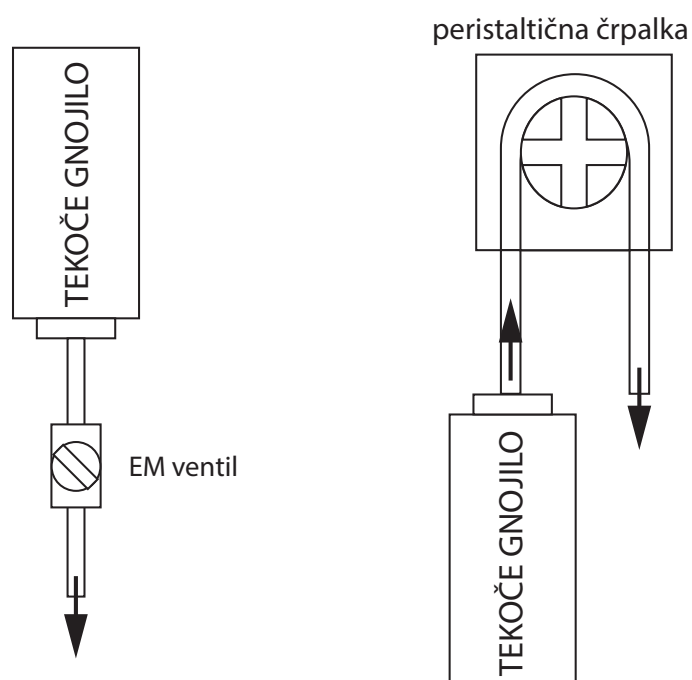
#### 4.2.4 Krmilna enota

Krmilna enota predstavlja glavni del celotnega sistema za nadzor in krmiljenje zelene stene in je edini del, brez katerega celoten sistem ne deluje. Med seboj povezuje vse ostale enote sistema. Skrbi za procesiranje vseh podatkov, ki jih ostale enote sistema pošiljajo, ter posreduje potrebne informacije enotam sistema, katerim so namenjene.

Krmilna enota je realizirana na razvojni ploščici Raspberry Pi.

#### 4.2.5 Sistem za nadzor in upravljanje

Sistem za nadzor in upravljanje je ločena enota sistema, ki jo lahko vključimo v sistem, lahko pa sistem deluje tudi brez nje. Gre za Django aplikacijo, ki jo s pomočjo Django strežnika lahko poženemo na samem Raspberry Pi-ju, ali pa jo postavimo na kakšen oddaljen strežnik. Slednja rešitev pride prav v primeru, ko je dostopov do sistema veliko (npr. v primeru več uporabnikov), ali pa želimo podatke o večjem številu sten zbirati na enem mestu.



Slika 4.4: Različna načina doziranja gnojila. Levi način prikazuje doziranje z uporabo elektromagnetnega ventila, desni način pa z uporabo peristaltične črpalke.



Sistem za nadzor in upravljanje je z Raspberry Pi-jem povezan preko posebnega sistema za komunikacijo. Ta skrbi za neposredno pošiljanje podatkov o meritvah, ter pridobivanje morebitnih podatkov iz sistema za nadzor in upravljanje. Vsa komunikacija poteka preko REST klicev.

## 4.3 Opis vezja

V tem poglavju je opisano vezje, ki ga je bilo potrebno dodatno zasnovati, da smo lahko na Raspberry Pi priklopili primerne senzorje in z njih prebrali podatke.

### 4.3.1 Priklop temperaturnega senzorja

Temperaturni senzor DS1820 za priklop na Raspberry Pi potrebuje zgolj en dodatni upor, ostale kontakte priklopimo neposredno na vhodne kanale Raspberry Pi-ja. Shema priklopa je prikazana na sliki 4.5.

### 4.3.2 Priklop senzorja vlažnosti

Priklop senzorja vlažnosti je malce bolj kompliciran, saj smo se odločili za senzor, ki meri zgolj prevodnost med dvema prevodnima členoma. Potrebujemo dva upora, katerih vrednosti pa moramo izbrati z eksperimentiranjem. Prevodnost substratov je namreč različna in s pravo izbiro uporov si zagotovimo kar se da veliko merilno območje. Shema priklopa prikazuje slika 4.5. Opazimo lahko, da senzor vlažnosti ni priklopljen neposredno na vhodne kanale Raspberry Pi-ja, temveč na analogno-digitalni pretvornik, saj Raspberry Pi sam po sebi nima analognih vhodno/izhodnih kanalov in jih je potrebno realizirati z dodatnim čipom.

### 4.3.3 Priklop EC senzorja

EC senzor, ki smo ga izbrali za implementacijo, je analogni in ga brez dodatnih elementov priklopimo neposredno na analogne vhodne kanale. Ker

Raspberry Pi takih kanalov nima, problem rešimo z dodatnim čipom, kot smo to storili v primeru senzorja vlažnosti. Shema priklopa je prikazana na sliki 4.5.

#### 4.3.4 Priklop vodne črpalke

V našem primeru izbrana vodna črpalka za svoje delovanje potrebuje napetost 12V, česar neposredno z uporabo izhodnih kanalov na Raspberry Pi-ju ne moremo doseči. Zato smo uporabili dodatno napajanje, ter rele, ki služi kot stikalo za vklop/izklop napajanja. Potrebno je poudariti, da izhodni kanali na Raspberry Pi-ju tokovno niso dovolj zmogljivi za preklp releja, kar pa enostavno rešimo tako, da na rele priklopimo dva izhodna kanala, ter z obema preklapljammo rele. Shema priklopa prikazuje slika 4.5.

#### 4.3.5 Priklop elektromagnetnih ventilov

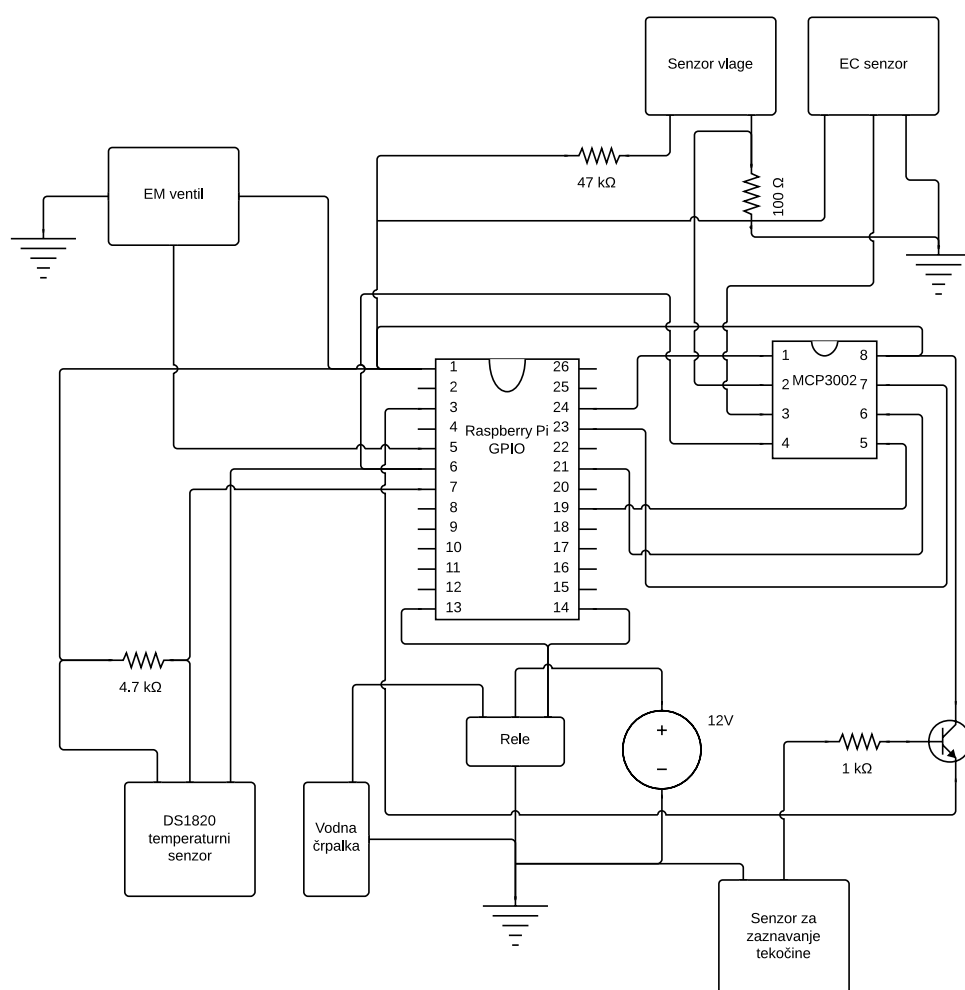
Priklop elektromagnetnih ventilov je najbolj enostaven, saj jih lahko priklopimo neposredno na izhodne kanale na Raspberry Pi-ju.

#### 4.3.6 Priklop senzorja za zaznavanje tekočine

Kot smo omenili v prejšnjem poglavju, je senzor za zaznavanje tekočine zelo enostaven. Tekočino zaznava z merjenjem prevodnosti. Ker pa so, kot že omenjeno, izhodni kanali na Raspberry Pi-ju tokovno prešibki, moramo uporabiti tranzistor, sicer tudi v primeru prevodnosti le-te ne zaznamo. Shema priklopa prikazuje slika 4.5.

#### 4.3.7 Shema celotnega vezja

Na sliki 4.5 je prikazana poenostavljena shema celotnega vezja.



Slika 4.5: Poenostavljena shema vezja za krmiljenje zelene stene.

## 4.4 Namestitev in zagon razvojne ploščice Raspberry Pi

V tem razdelku je opisan postopek namestitve in konfiguracije Raspberry Pi sistema, ter postopek zagona vseh potrebnih skript za delovanje celotnega sistema, vključno s spletnim uporabniškim vmesnikom za nadzor in upravljanje zelene stene.

### 4.4.1 Namestitev operacijskega sistema Raspbian

Namestitev operacijskega sistema na Raspberry Pi je precej enostavna. Na spletu lahko dobimo vnaprej pripravljeno sliko operacijskega sistema Raspbian, ki je posebej za Raspberry Pi prilagojena različica operacijskega sistema Debian.

Priprava SD spominske kartice:

1. S strani <http://www.raspberrypi.org/downloads> prenesemo zadnjo različico slike operacijskega sistema Raspbian.
2. Preneseno .zip datoteko razširimo. V njej se nahaja .img datoteka.
3. Z uporabo programa Win32DiskImager (windows uporabniki) .img datoteko zapišemo na SD spominsko kartico velikosti vsaj 4GB. Običajno je potrebno Win32DiskImager pognati kot administrator.

Tako pripravljena SD kartica sedaj vsebuje operacijski sistem Raspbian. Zaženemo ga enostavno tako, da SD kartico vstavimo v Raspberry Pi, ter ga priklopimo na napajanje [6].

Ob prvem zagonu izvedemo naslednje ukaze, da posodobimo sistem in namestimo pakete ki so nujni za nadaljnje delo:

```
sudo apt-get update  
sudo apt-get upgrade  
sudo apt-get install python-pip
```

```
pi@raspberrypi ~ $ ls /dev/spidev0.*  
/dev/spidev0.0 /dev/spidev0.1
```

Slika 4.6: Omogočen SPI vmesnik - v primeru da je SPI vmesnik omogočen, se v mapi nahajata dve datoteki, kot je prikazano na sliki.

#### 4.4.2 Konfiguracija Raspberry Pi-ja za delo z V/I napravami

Čeprav je Raspberry Pi zasnovan tako, da je delo z vhodno/izhodnimi napravami karseda enostavno, je ob prvem zagonu vseeno potrebno namestiti nekaj dodatnih paketov in spremeniti nekaj sistemskih nastavitev, preden sploh lahko komuniciramo z vhodno/izhodnimi napravami.

Za splošno branje/pisanje digitalnih pinov je potrebna knjižnica Rpi.GPIO. Preden se lotimo namestitve le-te pa potrebujemo orodje Python Development:

```
sudo apt-get install python-dev  
sudo apt-get install python-rpi.gpio
```

Za delovanje A/D pretvornika je v operacijskem sistemu potrebno omogočiti SPI vmesnik, preko katerega Raspberry Pi komunicira z A/D pretvornikom. Najprej moramo v operacijskem sistemu dovoliti uporabo SPI vmesnika, kar dosežemo tako, da v konfiguracijski datoteki `/etc/modprobe.d/raspi-blacklist.conf` zakomentiramo vrstico `blacklist spi-bcm2708`. Po tej operaciji je potreben ponoven zagon operacijskega sistema. Delovanje SPI vmesnika preverimo tako da izpišemo vsebino mape `/dev/`. V njej se ob pravilni konfiguraciji nahajata dve datoteki, kot prikazuje Slika 4.6

Za uporabo SPI vmesnika v Python skripti pa je potrebno namestiti še dodatno knjižnico:

```
sudo pip install spidev
```

```
pi@raspberrypi /sys/bus/w1/devices $ ls
10-000802842e2e w1_bus_master1
```

Slika 4.7: Priključena naprava DS1820 - če je temperaturni senzor priključen na sistem, med napravami opazimo napravo oblike 10-xxxx

Za delovanje temperaturnega senzorja DS1820 ki deluje na principu »1-wire« je potrebno v operacijskem sistemu omogočiti napravo. To storimo z naslednjimi ukazi:

```
sudo modprobe w1-gpio
sudo modprobe w1-therm
```

Po izvedbi zgornjih ukazov se med napravami v operacijskem sistemu (/sys/bus/w1/devices) nahaja nova naprava z imenom oblike 10-xxxx (glej sliko 4.7). To ime predstavlja unikatno serijsko številko naprave in jo bomo kasneje v skripti uporabili, ko bomo želeli brati vrednosti naprave. V primeru da na sistem priklopimo več takšnih senzorjev, bo v /sys/bus/w1/devices prikazanih več naprav, vsaka pa bo imela svojo unikatno serijsko številko.

### 4.4.3 Namestitev ostalih knjižnic

Celoten sistem nadzora in krmiljenja zelene stene poleg skript za krmiljenje samega strojnega (senzorskega) dela vključuje tudi ostale aplikacije, ki za svoje delovanje potrebujejo razne knjižnice, vendar se na tem mestu ne bomo podrobneje spuščali v namestitev uporabniškega vmesnika ter sistema za komunikacijo z uporabniškim vmesnikom.

Vse knjižnice, ki so potrebne za delovanje ostalih delov sistema so navedene v datoteki requirements.txt. Vse potrebne knjižnice lahko namestimo v enem koraku kar iz te datoteke:

```
sudo pip install -r requirements.txt
```

## 4.5 Zagonska skripta

Od sistema za nadzor ter avtomatsko krmiljenje zelene stene je pričakovano, da je sistem stabilen ter odporen na morebitne zunanje motnje. Ob izpadu električnega toka je tako potrebno zagotoviti da se ob ponovni prisotnosti električnega toka sistem povrne v prejšnje stanje.

Delno nam pri tem pomaga že sam Raspberry Pi, ki se ob prisotnosti napajanja avtomatsko prižge in zažene operacijski sistem. Od tu dalje pa moramo sami poskrbeti za morebitno inicializacijo senzorjev ter zagon skript. V ta namen lahko napišemo skripto v skriptnem jeziku lupine Bash, ji dodelimo pravico izvajanja ter jo postavimo v mapo `/etc/init.d/`. Po tem je potrebno skripto še registrirati ter znova zagnati Raspberry Pi, kar storimo z naslednjimi ukazi:

```
sudo chmod 755 /etc/init.d/greenwall
sudo update-rc.d greenwall defaults
```

Zagonska skripta je v našem primeru precej enostavna:

```
sudo modprobe w1-gpio
sudo modprobe w1-therm
python /home/pi/Desktop/GreenWallControl/greenwalldjango/manage.py \
runserver 0.0.0.0:8000 &
sudo python /home/pi/Desktop/GreenWallControl/start.py &
```

Prvi dve vrstici sta bili omenjeni že v prejšnjem poglavju in služita inicializaciji temperaturnega senzorja DS1820. Tretja vrstica požene Django strežnik, na katerem teče uporabniški vmesnik. Več o uporabniškem vmesniku bomo povedali v enem izmed naslednjih poglavij. Zadnja vrstica pa zažene skripto, ki skrbi za krmiljenje strojnega dela sistema, ter posredovanje informacij uporabniškemu vmesniku.

Razlog za zagon Django strežnika pred glavno skripto je zgolj v tem da želimo v Django aplikaciji beležiti vse podatke ki jih skripta pošilja, česar pa ne bi dosegli v primeru da bi Django strežnik zagnali po tem ko se skripta

že začne izvajati. Seveda pa lahko stvar izvedemo tudi na ta način in s tem ni nič narobe, le vseh podatkov, ki jih skripta pošilja, ne bomo uspeli zajeti.

## 4.6 Program za krmiljenje vezja

Glavni program, ki krmili celotno zgoraj opisano vezje, je napisan v programskem jeziku Python. Python skripta s programsko kodo se požene takoj ko je operacijski sistem pripravljen in se izvaja, dokler je Raspberry Pi vklopljen. Kot pri večini sistemov, ki uporabljajo mikrokrmilnike, je tudi tu program pravzaprav neskončna zanka, saj želimo doseči, da se izvaja, dokler je naprava vklopljena.

## 4.7 Komunikacija z uporabniškim vmesnikom

Za komunikacijo med programom za krmiljenje vezja in uporabniškim vmesnikom skrbi posebna skripta, katere funkcije proži program, ki krmili vezje. V vsakem obhodu neskončne zanke namreč preveri, ali v vrsti čaka kakšno sporočilo od uporabniškega vmesnika, hkrati pa posreduje podatke o prebranih meritvah uporabniškemu vmesniku.

Skripta za komunikacijo z uporabniškim vmesnikom vsebuje več metod, ki si jih bomo v nadaljevanju podrobneje ogledali.

`get_current_settings()` skrbi za pridobivanje aktualnih uporabnikovih nastavitev, nastavljenih preko uporabniškega vmesnika. Metodo kliče program za krmiljenje vezja vsakič, ko preverja, ali so prebrane vrednosti senzorjev znotraj območja zelenih nastavitev.

`get_irrigate_request()` je metoda ki preveri ali je uporabnik znotraj uporabniškega vmesnika sprožil zahtevo za ročno zalivanje. Poleg te metode imamo še podobno metodo `get_fertilize_request()`, ki pa se razlikuje le v tem da preverja zahteve za gnojenje.



Pri obeh metodah je priporočljiva uporaba destruktivnega branja - ko zahtevo preberemo, jo takoj pobrišemo.

`save_measurement(ec, temp, humidity)` posreduje podatke o meritvah uporabniškemu vmesniku. Trenutno metoda sprejme tri parametre:

- `konduktivnost(ec)`,
- `temperatura(temp)` ter
- `vlažnost(humidity)`,

po potrebi pa lahko dodamo dodatne parametre (v primeru dodatnih meritev).

`send_warning(text)` omogoča pošiljanje obvestil uporabniškemu vmesniku.

V parametru `text` podamo tekst, ki naj se izpiše ob obvestilu.

Vse metode za komunikacijo uporabljajo REST klice, s tem pa dosežemo da je sam uporabniški vmesnik kar se le da neodvisen od programa za krmljenje vezja. Lahko je napisan v praktično kateremkoli jeziku ter se nahaja kjerkoli, saj komunikacija pri REST klicih poteka preko protokola HTTP.

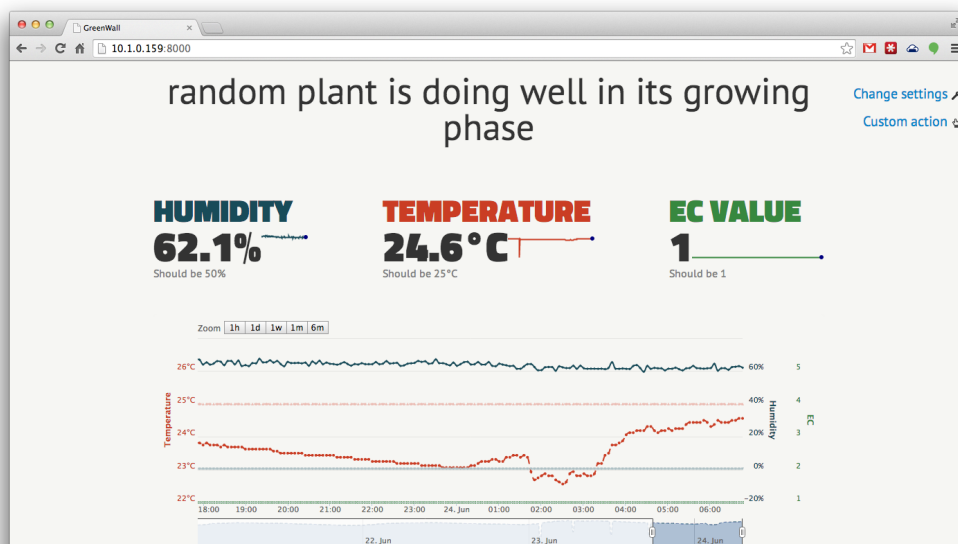
## 4.8 Uporabniški vmesnik in uporaba

Kot smo že omenili, je uporabniški vmesnik povsem neodvisen od programa za krmljenje vezja, prav tako pa imamo pri sami implementaciji uporabniškega vmesnika precej proste roke, saj je edina zahteva s strani programa za krmljenje vezja le to, da izbrana tehnologija omogoča komunikacijo preko REST klicev.

V našem primeru smo se odločili za uporabo Django aplikacije, spletni uporabniški vmesnik pa je narejen z uporabo tehnologij HTML5 ter JavaScript.

Uporabniški vmesnik prikazuje slika 4.8, ponuja pa nam naslednje:

**Prikaz trenutnih vrednosti temperature, vlažnosti ter EC**



Slika 4.8: Uporabniški vmesnik. Na interaktivnem grafu je možnost izbire časovnega obdobja meritev.

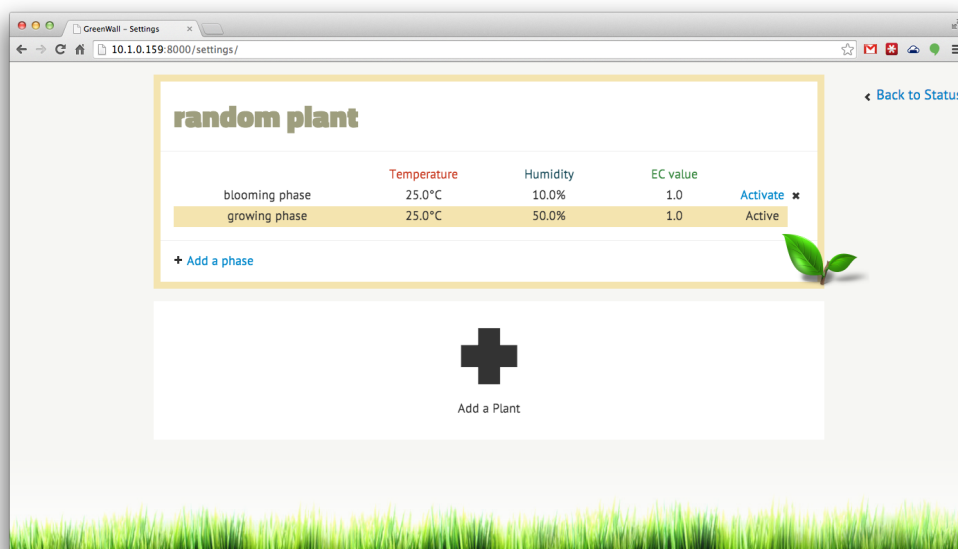
**Prikaz zgodovine meritev na interaktivnem grafu** Uporabnik ima možnost izbire časovnega obdobja meritev.

**Prikaz pomembnih obvestil** Takšno obvestilo je na primer prenizek nivo vode. Obvestila se pojavijo v zgornjem delu uporabniškega vmesnika.

**Izbira rastlin, ter faze rasti** Vsaka rastlina ima v različnih fazah rasti drugačne optimalne pogoje za rast. Izbiro rastline ter faze rasti izvedemo v nastavitvah (glej sliko 4.9).

**Možnost dodajanja novih rastlin** V nastavitvah imamo na dnu možnost dodajanja novih rastlin, kot to prikazuje slika 4.9. Pri dodajanju rastline izberemo optimalne parametre za rast.

**Ročno proženje zalivanja ter gnojenja** V meniju izberemo zalivanje oziroma gnojenje, ter nastavimo čas zalivanja, oziroma količino gnojila.



Slika 4.9: Nastavitve uporabniškega vmesnika. Slika prikazuje možnost izbire rastline, ter dodajanje novih rastlin.



## Poglavje 5

# Sklepne ugotovitve in nadaljnje delo

Tekom diplomskega dela smo izdelali prototip krmiljena in nadzora zelene stene, ki se je v praksi dobro obnesel. Končni produkt prikazuje slika 5.1. Delovanje sistema smo testirali več mesecev, v tem obdobju pa ugotovili kar nekaj manjših pomanjkljivosti v delovanju, ki pa smo jih sproti odpravili in v diplomskem delu niso opisane, pač pa je opisana zgolj končna rešitev. Tudi pri končni rešitvi pa smo našli nekaj problemov:

**Slabo delovanje EC senzorja** EC senzor, ki smo ga izbrali, ne deluje po pričakovanjih, tudi sicer pa je dokumentacija proizvajalca zelo skromna. Za nadaljnje delo bi bilo potrebno poiskati primernejši senzor.

**Oksidacija senzorja za vlago** Senzor za vlago, ki smo ga izdelali, je po določenem času začel oksidirati, saj se kljub vodoodpornosti samega materiala, zaradi stalne prisotnosti električnega toka pojavi proces elektrolize. Problem bi lahko rešili z menjavanjem smeri električnega toka pri vsaki meritvi.

**Neenakomerna vlažnost** Opazili smo, da je v zgornjem delu panela vlažnost vedno veliko nižja kot v spodnjem delu, žal pa je to problem same zasnove panela. Verjetno bi bilo primerneje, da bi bil panel razdeljen na



Slika 5.1: Delujoča zelena stena zasajena z začimbnicami in zelišči.

več horizontalnih plasti, ter bi merili vlažnost v vsaki plasti in zalivali vsako plast ločeno.

Celoten sistem bi bilo pametno razširiti še z obdelavo podatkov o meritvah v oblaku, ter izdelati priporočilni sistem, kot je bilo predvideno pri zasnovi sistema, vendar pa do implementacije ni prišlo. To bi bil vsekakor velik doprinos k obstoječem sistemu, saj bi uporabniku poenostavili oziroma povsem avtomatizirali izbiro najbolj optimalnih nastavitev.





# Literatura

- [1] S. Loh, "Living Walls - a Way to Green the Built Environment", *BEDP Environment Design Guide*, ACT, Australia, str. 1-7, 2008. Dostopno na:  
<http://math.unife.it/lm.ecologia/Insegnamenti/management-degli-ecosistemi/materiale-didattico/Loh%202008%20living%20walls.pdf>
- [2] K. Perini, M. Ottele, E. M. Haas, R. Raiteri "Vertical greening systems, a process tree for green façades and living walls", *Urban Ecosystems*, Springer US, št. 16, str. 265-277, 2013. Dostopno na:  
<http://link.springer.com/article/10.1007%2Fs11252-012-0262-3>
- [3] R. A. Francis, J. Lorimer "Urban reconciliation ecology: The potential of living roofs and walls", *Journal of Environmental Management*, Elsevier, št. 92, zv. 6, str. 1429-1437, Junij 2011. Dostopno na:  
<http://www.sciencedirect.com/science/article/pii/S0301479711000132>
- [4] Raspberry Pi. Dostopno na:  
<http://www.raspberrypi.org>
- [5] What is a Raspberry Pi. Dostopno na:  
<http://www.raspberrypi.org/help/what-is-a-raspberry-pi/>
- [6] Raspberry Pi Installation. Dostopno na:  
<http://www.raspberrypi.org/documentation/installation/>
- [7] DS1820 specifikacija. Dostopno na:  
<http://datasheets.maximintegrated.com/en/ds/DS18S20.pdf>

- [8] LFS1K0 specifikacija. Dostopno na:  
<http://www.farnell.com/datasheets/1644000.pdf>
- [9] Zelene stene podjetja Humko. Dostopno na:  
<http://www.greenwalls.si/>
- [10] Zelene stene podjetja Gsky. Dostopno na:  
<http://gsky.com/>
- [11] Zelene stene podjetja Ambius. Dostopno na:  
<http://www.ambius.com/>
- [12] Zelene stene podjetja Bin Fen. Dostopno na:  
<http://www.binfengreenwallsystem.com/>